3-12-2007

Anti-Forensics and the Digital Investigator

Gary C. Kessler
Champlain College

Recommended Citation

Kessler, G. C. (2007). Anti-Forensics and the Digital Investigator. DOI: https://doi.org/10.4225/75/57ad39ee7ff25

DOI: 10.4225/75/57ad39ee7ff25
5th Australian Digital Forensics Conference, Edith Cowan University, Perth Western Australia, December 3rd 2007.

Anti-Forensics and the Digital Investigator

Gary C. Kessler
Champlain College
Burlington, VT, USA
gary.kessler@champlain.edu

Edith Cowan University
Mount Lawley, WA, Australia

Abstract

Viewed generically, anti-forensics (AF) is that set of tactics and measures taken by someone who wants to thwart the digital investigation process. This paper describes some of the many AF tools and methods, under the broad classifications of data hiding, artefact wiping, trail obfuscation, and attacks on the forensics tools themselves. The concept of AF is neither new nor solely intended to be used by the criminal class; it also has legitimate use by those who wish to protect their privacy. This paper also introduces the concept of time-sensitive anti-forensics*, noting that AF procedures might be employed for the sole purpose of delaying rather than totally preventing the discovery of digital information.*

Keywords

Anti-forensics, data hiding, artefact wiping, trail obfuscation, attacks on computer forensics tools, privacy

INTRODUCING ANTI-FORENSICS

The term *anti-forensics* (AF) has recently entered into the vernacular of digital investigators. Although conceptually not new, it is instructive to observe that there is no clear industry definition (Harris, 2006). Rogers (2006), a practicing digital forensics educator and investigator, defines AF as "attempts to negatively affect the existence, amount, and/or quality of evidence from a crime scene, or make the examination of evidence difficult or impossible to conduct." Liu and Brown (2006), practicing creators of AF methods and tools, offer a slightly darker definition: "application of the scientific method to digital media in order to invalidate factual information for judicial review."

The term *forensics* is significant and quite specific -- whatever AF is pertains to the scientific analysis of evidence for court. Anti-forensics, then, is that set of tools, methods, and processes that hinder such analysis.

It is difficult to think of any *legitimate* uses of AF processes and tools. Indeed, most -- if not all -- digital investigators find AF to be the bane of their existence and only used by someone who has something to hide.

But AF might also be employed by the person who just wants to be left alone. An early text about computer forensics devoted significant time to the examination process as well as ways to thwart that process, all in the name of privacy (Caloyannides, 2001). One can argue that it is a fine line between protecting one's privacy and preventing a court-sanctioned search, but that line has existed for centuries -- only with digital devices does a letter that the writer burned well in the past continue to hang around on a hard drive or backup tape. And there are those that will argue that AF techniques can protect a Good Person from a Bad Government. Of course, those same tools can block a Good Government from investigating a Bad Person and that, of course, is the rub.

Laws, traditions, mores, and culture affect a society's view of privacy. Many nations purport to value people's privacy on the one hand but also recognize that the "right to privacy" -- should that right exist in a particular jurisdiction -- is not absolute.

CATEGORIES OF ANTI-FORENSICS METHODS

Much of the recent discussion in articles, conferences, and blogs seems to suggest that AF tools and methods have appeared suddenly (Harris, 2006; Liu & Brown, 2006; Rogers, 2006). While this is certainly true in some cases, it is also important to note that many forms of AF -- although not created to hinder the detection of evidence, per se -- have been around for quite some time.

Rogers (2006) suggests that there are four basic categories of AF: data hiding, artefact wiping, trail obfuscation, and attacks against the computer forensics process or tools. This section will provide examples of tools and processes that fit in these categories.

Data Hiding

Data hiding can be accomplished in a variety of ways. Hidden writing (i.e., *steganography*) has been around for over two thousand years. Digital steganography tools have been available since at least the mid-1990s and stego software is available for almost every computer operating system. *Any* form of digital information can be stored inside many types of carrier files, including image, audio, video, and executable files (StegoArchive.com, 2005).

Low-technology stego methods can also be employed to hinder computer forensics. As an example, a person can hide a picture, table, or text block under an image in a PowerPoint or Impress graphical presentation. Alternatively, a white text block over a white background can store a hidden message. Morse code messages can be embedded in a picture. Null ciphers form messages by selecting a pre-determined pattern of letters from a sequence of words. Many other forms of low-tech stego can be employed that no automated tool can detect (Kessler, 2004).

Covert channels in data communications protocols allow hidden communication over public and private networks. The Transmission Control Protocol/Internet Protocol (TCP/IP) suite, for example, has several weaknesses that can be exploited to allow covert communications. The concept of covert channels in networking protocols dates back at least 30 years (Ahsan, 2002; Rowland, 1996).

There are numerous ways to hide data, at least from cursory searches. Data can be hidden in the slack and unallocated spaces on computer hard drives, as well as the metadata of many types of files. The Master Boot Record (MBR), allocated but unused device driver registers and tables, protected system areas, and hidden (and possibly encrypted) partitions on hard drives are other places to secret information, as well as the Basic Input/Output System (BIOS) chip itself (Budimir & Slay, 2007). Homographic file names (where non-Latin characters that appear similar to Latin characters are used in the name), use of very long path names (greater than 256 characters), and use of hidden characters (e.g., 0xFF) in file names can all be used to hide data from the operating system. Data can also be hidden in closed sessions on compact discs, in another user's disk space on a poorly secured server, or out in the open using public shares. All of this data can still be found by forensics tools and the astute examiner, but they are harder to find and harder to explain to the non-technical audience.

Artefact Wiping

Artefact wiping tools have been available for many years. Wiping programs such as BC Wipe, Eraser, and PGP Wipe destroy data files using multiple overwrites that makes any retrievable impractical, if not impossible.

Automated artefact wiping software is available, ostensibly, to allow a user to recover storage space -- and protect one's privacy -- by removing unneeded temporary files that clutter up the hard drive. But software such as Evidence Eliminator, Secure Clean, and Window Washer remove browser history and cache files, delete certain operating system files, and wipe slack and unallocated space. Many of these programs come preconfigured to eliminate the detritus of various operating systems and common utilities (e.g., Windows, MS Office, Internet Explorer, AOL Instant Messenger, and Firefox), and have additional plug-ins for a large number of other common applications (e.g., Eudora, Picasa, RealAudio, and WinZip). Many guides are available that directly address which files should be cleaned out so as to make analysis difficult or to render forensics largely moot (Caloyannides, 2001; Geiger & Cranor, 2006).

The best of these tools do not, of course, merely delete offensive files -- which would be the benign approach -- but wipe them. Deleting unnecessary files is sufficient to recover disk space; wiping the files certainly does suggest that someone is interested in more than simple space recovery.

Artefact wiping tools make analysis more difficult for forensics examiners but the fact is that they are not perfect. Most of the programs leave identifiable traces of the wiping and many are not as complete as they advertise to be, often leaving behind remnants of the very things they are supposed to delete (Geiger & Cranor, 2006).

Trail Obfuscation

Trail obfuscation has been an issue since the 1970s with logon spoofing followed in the 1980s with IP and Medium Access Control (MAC) address spoofing. Denial-of-service (DoS) and distributed DoS attacks -- widespread since 1996 -- depend upon successful IP address spoofing and have made network intrusions more difficult to investigate. Indeed, defences to DoS/DDoS attacks have been more about prevention, response, and recovery rather than detection of the attacker, although some methods of IP traceback have been devised in order to track packets on the network back to their source (Levine & Kessler, 2002). So-called *onion routing*, however, can make network traffic analysis nearly impossible (Forte, 2002).

There are a variety of ways to confuse e-mail investigations. E-mail anonymizers ostensibly provide privacy services, preventing an investigator from determining the source of a sent mail message. Planting false headers, open Simple Mail Transfer Protocol (SMTP) proxies, and anonymous Secure Shell (SSH) tunnel servers are among the other mechanisms that can add complexity to tracking back the origins of e-mail. Web anonymizers hide a Web site user's identity and anonymity toolsets such as Tor can effectively bring an Internet-based investigation to a halt (Akin, 2003; EFF, 2007).

Trail obfuscation can also be accomplished by wiping and/or altering server log files and/or system event files, or altering the dates of various files (e.g., using `touch`). A Bad Guy hacker who can break into a server very likely has the knowledge to hide his/her tracks and/or leave false clues by modifying these files.

Attacks Against Computer Forensics Tools

Direct attacks on the computer forensics process are the newest type of AF and potentially the most threatening. Palmer (2001) describes six phases in the process of digital forensics; all are open to attack:

1. *Identification* refers to the method by which an investigator learns that there is some incident to investigate. This phase can be undermined by obscuring the incident, or hiding the nexus between the digital device and the event under investigation.
2. *Preservation* describes the steps by which the integrity of the evidence is maintained. This phase can be undermined by interrupting the evidentiary chain or calling into doubt the integrity of the evidence itself.
3. *Collection* is the process by which data from the evidence medium is acquired. This step can be undermined by limiting the completeness of the data being collected or calling into question the hardware, software, policies, and procedures by which evidence is gathered.
4. *Examination* addresses how the evidence data is viewed. This part of the process can be undermined by showing that the tools themselves are inadequate, incomplete, or otherwise not scientifically valid.
5. *Analysis* is the means by which an investigator draws conclusions from the evidence. This phase relies on the tools, investigative prowess of the examiner, and the rest of the evidence that was found. If a case hinges solely on digital evidence, the interpretation of the evidence is the part most open to attack.
6. *Presentation* refers to the methods by which the results of the digital investigation are presented to the court, jury, or other fact-finders. If the evidence is otherwise solid, anti-forensics tools and methods will be used to attack the reliability and thoroughness of the reports -- or the examiner.

Courts throughout the world have long had to deal with scientific evidence and have had to establish rules for what is acceptable and unacceptable in this realm. In the U.S., the guiding principle in federal courts and many state courts is patterned after the seminal case of Daubert v. Merrell Dow Pharmaceuticals (Supreme Court of the United States, 1993). According to *Daubert*, a judge can determine the admissibility of scientific evidence based upon four factors:

- *Testing:* Can -- and has -- the procedure been tested?
- *Error Rate:* Is there a known error rate of the procedure?
- *Publication:* Has the procedure been published and subject to peer review?
- *Acceptance:* Is the procedure generally accepted in the relevant scientific community?

Anti-forensics procedures, then, can attack the reliability of digital evidence; if the reliability of the evidence can be successfully called into question, it becomes worthless in a court of law. In fact, Van Buskirk and Liu (2006) argue that forensics software seems to have been granted a presumption of reliability by the courts that may be undeserved -- and actually in conflict with *Daubert*.

There have already been successful attacks on many of the major computer forensics tools, including EnCase, FTK, iLook, SleuthKit, and WinHex. Programs that fiddle with FAT directories, NTFS master file tables, and ext inodes have been around for years, as well as programs that write to file slack, alter file signatures, and flip bits in order to evade hashset detection (Rogers, 2006).

An example of the tension between the AF community and software providers is exemplified by a report from the 2007 U.S. Black Hat conference. In this instance, a group presented the results of applying several exploitation techniques to a number of commercial and open-source computer forensics applications (Guidance Software, 2007; Palmer, Newsham, Stamos, & Ridder, 2007). They concluded that:

- Forensics software vendors do not design their products to function in a hostile environment; i.e., they do not generally create software that could acquire evidence from machines that have been configured to withstand or falsify evidence during acquisition by known forensic tools.

- Forensics software developers do not create products that are properly protected against such flaws as stack overflows, improper management of memory pages, and unsafe exception handling leakage.
- Forensics software users do not apply sufficiently strong criteria to the evaluation of the products that they purchase. In fact, most computer forensics labs purchase the software that "everyone else" is using and do not perform independent tests of reliability, thoroughness, and veracity, particularly as new versions of the software get released.

If the aim of anti-forensics is to render moot digital evidence, then calling into question the effectiveness of the very tools that we use to find, analyse, examine, and report on this evidence will have a chilling effect on the entire digital investigation community. In the final analysis, the computer examiner may find all of the evidence that is present and interpret it correctly -- but if it is not believed in court, then the entire exercise is meaningless.

ADDITIONAL ASPECTS OF ANTI-FORENSICS

The Metasploit Project

The Metasploit Project is an open-source collaborative with a stated goal of providing information to the penetration testing, intrusion detection system (IDS) signature development, and information system exploit research communities (Metasploit LLC, 2007b). The Metasploit Anti-Forensics Project (Metasploit LLC, 2007a), in particular, has the stated goal of investigating the shortcomings in computer forensics tools, improving the digital forensics process, and validating forensics tools and processes. One output of this project has been the Metasploit Anti-Forensic Investigation Arsenal (MAFIA), a suite of programs that includes:

- *Sam Juicer* -- A program that acquires the hashes from the NT Security Access Manager (SAM) file without touching the hard drive
- *Slacker* -- A program to hide files within the slack space of NTFS files
- *Transmogrify* -- "Defeats" EnCase's file signature detection capabilities by allowing a user to mask and unmask files as any file type
- *Timestomp* -- A program that alters all four NT File System (NTFS) file times: modified, access, creation, and file entry update (so-called MACE times)

These tools represent a combination of a demonstration of capability as much as practical ways in which a user can confuse digital forensics examinations; the software authors acknowledge that the software does not necessarily take the AF process to the n-th degree. Timestomp, for example, modifies only the MACE times stored in a file's $STANDARD_INFORMATION attribute and those not in the $FILE_NAME attribute, thus, leaving some indicator of suspicious activity. It is only a minor modification to the program, however, to make them more thorough (Liu & Brown, 2006).

While the MAFIA tools are successfully proving their point, they can also be used for other purposes -- such as hiding a guilty party's incriminating evidence or placing incriminating evidence on the drive of an innocent party.

Cryptography

Cryptography, in some ways, is the ultimate anti-forensics tool. And, of course, it is not new -- crypto tools have made, and will continue to make, digital investigations difficult or impossible. Cryptography is perhaps the most troublesome AF tool because it is easy for the general user to employ and, once turned on, required minimal maintenance or attention on the part of the user. That said, many applications use weak encryption which can easily be defeated (e.g., Wired Equivalent Privacy [WEP]) or users may not manage their keys well.

Crypto protection comes in a variety of flavours. Many applications, such as Adobe Acrobat, MS Office, and WinZIP, provide mechanisms so that users can password protect and/or encrypt individual files. Pretty Good Privacy (PGP) has been encrypting e-mails since the early-1990s. Encrypting file systems and whole disk encryption (e.g., PGP Desktop, SafeBoot, Vista with BitLocker, and Windows EFS) will continue to thwart -- or, at least, resist -- computer forensics examinations. Encrypted Internet-based communication (e.g., Secure Sockets Layer, Transaction Layer Security, virtual private networks, and IEEE 802.11 secure wireless networks) can make analysis of the contents of network traffic nearly impossible (Casey, 2004).

The User

One would assume that there is a cadre of users who will employ every tool in the arsenal to make computer examinations difficult. In general, there is a linear relationship between the difficulty in using an AF tool and how much the user really has to hide. As it is today,

- Not every user is going to install AF tools
- Not every user who installs AF tools will use them consistently, thus leaving behind usable information
- Not every user who uses AF tool will use them correctly, which will leave usable information
- Not all AF tools work as perfectly as advertised, thus leaving remnants and traces

TIME-SENSITIVE ANTI-FORENSICS

Another purpose for anti-forensics may be to "protect" certain data until it is moot. Rather than prevent forensics analysis from occurring, it may be sufficient to bog the examination process down until the data loses its evidentiary -- or intelligence -- value.

This is conceptually similar to the information security concept of *time-based security* (Schwartau, 1999). That model suggests that an information security system does not need to keep an attacker out forever but only long enough for the attack to be detected and for a response to be mounted. Expressed as a formula:

$$PS_t > D_t + R_t$$

where PS_t = the length of time that the security system can protect against an attack; D_t = the length of time to detect an attack; and R_t = the length of time to respond to an attack.

The user of anti-forensics methods wants to keep the digital investigator at bay for some period of time (possibly forever). As a formula, this might be expressed:

$$PAF_t > I_t + AQ_t + E_t + AN_t$$

where PAF_t = the length of time that the anti-forensics method can protect data against discovery; I_t = the length of time to identify potentially useful data; AQ_t = the length of time to acquire the data; E_t = the length of time to examine the data; and AN_t = the length of time to analyse the data.

In the security model, detection and response to an attack is usually automated and requires a relatively short period of time (i.e., seconds, minutes, or hours). Conversely, identification, acquisition, examination, and analysis of digital evidence require human intelligence and, generally, a relatively long period of time (i.e., days or weeks) even in the best of circumstances. In some cases the use of AF methods only adds a little to an otherwise lengthy time delay but in other cases can frustrate an active, time-sensitive investigation.

It is worth considering the notion that some class of users actually *wants* digital examinations to take place but to be slowed down, wasting the time, money, and personnel resources of the investigating agency. This class of user does not want to prevent the exam, per se; if so, they might just use 256-bit whole disk encryption and be done with it. Instead, they want the exams to take place, but they want them to take a long time. In some sense, they are trying to confuse *their* adversary by inundating them with information but keeping it at arm's length.

CONCLUSION

This paper has attempted to broaden the scope of those methods and processes that could be fairly described as *anti-forensics*. It has identified many -- certainly not all -- AF processes and further work is necessary to categorize these methods based upon their ease (and, therefore, likelihood) of use, efficacy, degree of difficulty to detect and/or overcome, and other characteristics.

Every attempt to make the digital forensics process harder has been met with changes in the process to address the new challenges. Encryption taught investigators to think twice before the pulling the plug on a computer; live imaging of both hard drives and RAM is becoming much more common in the field today. Steganography taught investigators to not take everything at face value. Malware taught investigators that the scene of the computer exam might be hostile. AF software that attacks our tools may or may not result in better tools but will certainly cause us to change the process so as to rely less on fully automated exams; more human intelligence -- and time -- will be needed for investigations and this will also demand more detailed knowledge of file systems.

It is important to note that while many of the AF methods might make information derived from an investigation useless as evidence in court, they may not diminish the intelligence value of the information; *reasonable doubt* in the mind of a jury does not translate into non-actionable information for an intelligence gatherer. Other AF methods, of course, do secure data from the digital investigation process although it is unclear where the crossover of the value of the information retrieved and the "cost" of the resources expended to retrieve the data occurs.

Anti-forensics tools and methods will continue to provide difficulties and challenges to the digital investigation and e-discovery communities. As AF developers continue to produce tools, however, it becomes incumbent upon academia and industry to coordinate and fund anti-AF research and development. This may well be the next New Thing in digital investigations.

ACKNOWLEDGEMENTS

The author is partially supported by Grant No. 2006-DD-BX-0282 awarded by the Bureau of Justice Assistance. The Bureau of Justice Assistance is a component of the Office of Justice Programs, which also includes the Bureau of Justice Statistics, the National Institute of Justice, the Office of Juvenile Justice and Delinquency Prevention, and the Office for Victims of Crime. Points of view or opinions in this document are those of the author and do not represent the official position or policies of the United State Department of Justice.

ABOUT THE AUTHOR

Gary C. Kessler, Ed.S., CCE, CISSP is director of the Champlain College Center for Digital Investigation (C3DI) and an associate professor and director of the Computer & Digital Forensics program at Champlain College in Burlington, Vermont, and an adjunct associate professor at Edith Cowan University in Mount Lawley, Western Australia. He is a member of the High Technology Crime Investigation Association (HTCIA) and International Society of Forensic Computer Examiners (ISFCE). Kessler is also a technical adviser to the Vermont Internet Crimes Against Children (ICAC) and Internet Crimes Task Forces, a member of the editorial board of the *Journal of Digital Forensics, Security and Law (JDFSL)*, an associate editor of the *Journal of Digital Forensic Practice (JDFP)*, and a principal in GKS Digital Services, LLC.

REFERENCES

Ahsan, K. (2002). *Covert Channel Analysis and Data Hiding in TCP/IP*. Thesis presented to the Edwards S. Rogers Sr. Graduate Department of Electrical and Computer Engineering, University of Toronto, Toronto, Ontario. Retrieved September 11, 2007, from http://gray-world.net/papers/ahsan02.pdf

Akin, T. (2003). WebMail Forensics. BlackHat Briefings. Retrieved September 11, 2007, from http://www.blackhat.com/presentations/bh-usa-03/bh-us-03-akin.pdf

Budimir, N., & Slay, J. (2007). Identifying Non-Volatile Data Storage Areas: Unique Notebook Identification Information as Digital Evidence. *Journal of Digital Forensics, Security and Law*, *2*(1), 75-91.

Caloyannides, M.A. (2001). *Computer Forensics and Privacy*. Boston: Artech House.

Casey, E. (2004). *Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet* (2nd ed.). London: Elsevier Academic Press.

Electronic Frontier Foundation (EFF). (2007, September 18). Tor Web page. Retrieved September 18, 2007, from http://tor.eff.org/index.html.en

Forte, D. (2002, August). Analyzing the Difficulties in Backtracking the Onion Router's Traffic. *Proceedings of the 2002 Digital Forensics Research Workshop*. Retrieved September 11, 2007, from https://www.dfrws.org/2002/papers/Papers/Dario_Forte.pdf

Geiger, M., & Cranor, L.F. (2006, September/October). Scrubbing Stubborn Data: An Evaluation of Counter-Forensic Privacy Tools. *IEEE Security & Privacy*, *4*(5), 16-25.

Guidance Software. (2007, July 26). Guidance Software Response to iSEC Report. Retrieved September 11, 2007, from http://www.securityfocus.com/archive/1/474727

Harris, R. (2006). Arriving at an Anti-Forensics Consensus: Examining How to Define and Control the Anti-Forensics Problem. *Proceedings of the 2006 Digital Forensics Research Workshop*. *Digital Investigation*, *3*(S), S44-S49. Retrieved September 11, 2007, from http://dfrws.org/2006/proceedings/6-Harris.pdf

Kessler, G.C. (2004, July). An Overview of Steganography for the Computer Forensics Examiner. *Forensics Science Communication*, *6*(3). Retrieved September 11, 2007, from http://www.fbi.gov/hq/lab/fsc /backissu/july2004/research/2004_03_research01.htm

Levine, D.E., & Kessler, G.C. (2002). Denial of Service Attacks. In M. Kabay & S. Bosworth (Eds.), *Computer Security Handbook*, 4th ed. New York: John Wiley & Sons.

Liu, V., & Brown, F. (2006, April 3). Bleeding-Edge Anti-Forensics. Presentation at InfoSec World 2006. Retrieved September 11, 2007, from stachliu.com/files/InfoSecWorld_2006-K2-Bleeding_Edge_AntiForensics.ppt

Metasploit LLC. (2007a). Metasploit Anti-forensics home page. Retrieved September 11, 2007, from http://www.metasploit.com/projects/antiforensics/)

Metasploit LLC. (2007b). Metasploit Project home page. Retrieved September 11, 2007, from http://www.metasploit.com/

Palmer, C., Newsham, T., Stamos, A., & Ridder, C. (2007, August 1). Breaking Forensics Software: Weaknesses in Critical Evidence Collection. Abstract of presentation at Black Hat USA 2007. Retrieved September 11, 2007, from http://www.blackhat.com/html/bh-usa-07/bh-usa-07-speakers.html#Palmer

Palmer, G. (2001, November 6). *A Road Map for Digital Forensics Research*. Digital Forensic Research Workshop (DFRWS) Technical Report (DTR) T001-01 Final. Retrieved September 11, 2007, from http://www.dfrws.org/2001/dfrws-rm-final.pdf

Rogers, M. (2006, March 22). Panel session at CERIAS 2006 Information Security Symposium. Retrieved September 11, 2007, from http://www.cerias.purdue.edu/symposium/2006/materials/pdfs/antiforensics.pdf

Rowland, C.H. (1997, May 5). Covert Channels in the TCP/IP Protocol Suite. *First Monday*, *2*(5). Retrieved September 11, 2007, from http://www.firstmonday.org/issues/issue2_5/rowland/

Schwartau, W. (1999). *Time Based Security*. Seminole, FL: Interpact Press.

StegoArchive.com. (2005). Stego Archive Web site. Retrieved September 11, 2007, from http://www.stegoarchive.com

Supreme Court of the United States. (1993). *Daubert v. Merrell Dow Pharmaceuticals* (92-102), 509 U.S. 579. Retrieved September 11, 2007, from http://supct.law.cornell.edu/supct/html/92-102.ZS.html

Van Buskirk, E., & Liu, V.T. (2006, March). Digital Evidence: Challenging the Presumption of Reliability. *Journal of Digital Forensic Practice*, *1*(1), 19-26.

COPYRIGHT